REMARKS

Claims 1-18 were pending as of the data of the last Official Action (June 9, 2005).

Claims 1 and 13 have been amended to insert a comma (",") before "said correspondences" in each claim, and claim 12 has been amended to insert "from" before "Unisys," all to correct obvious minor informalities.

Rejections Under 35 U.S.C. § 103(a)

Claims 1, 3-11 and 13-18 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Nishiyama et al. (5,859,977) in view of Wischinski (6,801,920). Claims 2 and 12 stand rejected under Section 103(a) as being unpatentable over Nishiyama and Wischinski and further in view of Bryan et al. (6,591,418) or Mutschler et al (5,974,430), respectively.

With respect to the independent claims 1, 13 and 14, the Official Action essentially asserts that Nishiyama teaches all of the features of the claims, except for a "network listening program (web server)." Official Action, ¶ 4. The Official Action further asserts that Wischinski teaches the claimed "network listening program" and that it would have been obvious to combine that teaching with Nishiyama to produce the claimed invention. The applicants respectfully disagree and submit that the system of Nishiyama is quite different from what the present application claims.

The Claimed Invention

The present invention allows a server computer to run different versions of a software application program in such a way that clients of the server can issue requests for service and have those requests serviced by a selected one of the versions of the application program running on the server. *See* Summary of the Invention, pp. 2-3 ("[a] plurality of versions of software application programs can be handled by a single server serving multiple user-clients who each need access to specific ones of the plurality of versions"). One advantageous use of the claimed invention is during an upgrade from a current version of a software application program to a new version. With the invention, both the old version of the software application program and the new version of the software application program can run simultaneously on a server. A client who still needs to access the old version of the software,

because for example certain "legacy" functionality may not be available in the new version, can issue a request to the server that specifies that the request be routed to and handled by the old version of the software application program. On the other hand, a client that requires a feature that is only available in the new version of the software can issue a request to the server that specifies that the request be routed to and handled by the new version of the software running on the server. *See, e.g.,* Specification, p. 4.

According to the invention, each version of a software application program running on a server (and its associated data) are referred to "logically" as a "site":

> A site is a *logical* combination of the version of the software 96 and any data 97 required for the software application program 96 to run for this client and service the request.

Specification, p. 9 (emphasis added). Each "site" is then assigned a "SiteID," and a table is maintained on the server that associates each SiteID with its corresponding site (*i.e.,* the corresponding version of the software program running on the server). When a client issues a request to the server, the request may include the SiteID associated with a particular version of the software application program that the client wants to handle the request. In one embodiment, a "network listening program" on the server receives the request, and an "access control manager" determines from the table (and the received SiteID) to which version of the software application program the request should be routed (*i.e.,* linked). *See,* Specification, pp. 4-5. The request is then routed to the identified version, and that version of the software processes the request and returns any response to the client. These features are reflected in each of the independent claims 1, 13 and 14.

For example, independent claim 14 recites that "multiple versions of [a] software application program are maintained for servicing requests on a server" and that the following steps are performed:

> receiving a user request at a server,
>
> reading a SiteID code identifying a user site from within said user request,
>
> determining with reference to a table which one of a plurality of versions of a software application program on said server is indicated by said request,
>
> linking said request to said one version.

Independent claims 1 and 13 recite similar features in "system" claim form. The applicants respectfully submit that neither Nishiyama, Wischinski, Bryan, Mutschler or any other cited art of record, alone or in combination, teaches these features of the claimed invention.

The Claims Are Patentable Over The Cited Art

Nishiyama describes a system that manages the distribution of software to various *physical* sites (*i.e.,* computers) in a network. A "software distribution management table" is used to keep track of which versions of various software have been distributed to a given physical site and also the particular method used to distribute the software to that site (*e.g.,* bulletin board, broadcast or download). The purpose of the system is to manage software upgrades throughout the network. Specifically, as explained in Nishiyama,

> The software distribution management table 301 ... is a table for *managing software and its version and revision distributed to respective sites.*
>
> In a site name field 302, site names managed by the software distribution management table 301 are stored. In a software name field 303, names of all softwares in the system managed by the software distribution management table 301 are stored as "software-1, software-2, . . . ". This software name field 303 is further divided into two parts, i.e., a management kind field 304 and a version number field 305. The management kind of the above described software and the identification information of matter to be distributed are stored in the management kind field 304. *As for the management kind, the above described bulletin board method, broad cast method, and down load method are denoted by 1, 2 and 3, respectively.* As for the matter to be distributed, source and program (having execute form) are denoted by S and P, respectively. . . .
>
> At the time of distribution, the version number of the software which has been distributed is stored in the version number field 305 of the pertinent software, i.e., the software-1 in case of FIG. 3, in the software distribution management table 301.
>
> *By using the method heretofore described, version/revision management of software of the information system becomes possible.*

Col. 7, ln. 44 – col. 8, ln. 21 (emphasis added). Nishiyama's software version/revision management system has nothing to do with the applicants' claimed system and method for servicing of user requests by different versions of a software application program running on a server.

The Official Action incorrectly asserts that column 5, lines 4-12 and column 8, lines 36-52 of Nishiyama teach receiving a user request that includes a SiteID and assigning that request to a particular version of a software application program in order to have that version of the software service the request. *See*, Official Action, ¶¶ 4 and 25. On the contrary, those sections of Nishiyama merely describe the following:

> The computer system of FIG. 1 includes the software maintenance system 101 for exercising centralized management of software of the entire system, the information system 102 for performing the processing of the information system, the control system 103 for performing the processing of the control system, the mixed information and control system 104 for performing mixed processing of the information system and the control system, and the network 105 for connecting those systems. (col. 5, lines 4-12)

and

> (a) Site management function
>
> Each of computers such as PC, PRC and WS included in the system is called site. One site is provided with one name. This is called site name and used to identify the site. In the site management function of the present embodiment, registration and deletion of site names are managed. In the site management function of the present embodiment, program attributes such as name of software stored in each site, version and revision, occupation size in the memory, and resident (which means that the program is stored on the main memory) or nonresident (which means the program is read from an external memory such as a disk every time the program is started) are also managed. *In the network computer system, the site management function makes it possible to immediately discriminate the version and revision of each site and thereby prevent a mistake at the time of program replacement.* (col. 8, lines 36-52)(emphasis added).

Nowhere in these cited portions does the Nishiyama reference teach or suggest receiving a user request that includes a SiteID and routing or linking that request to a particular version of a software application program in order to have that version of the software service the request. On the contrary, these portions of Nishiyama reinforce that the system of Nishiyama is merely a software version/revision management system. Indeed, as the second paragraph above concludes, "the site management function makes it possible to immediately discriminate the *version and revision of each site* and thereby *prevent a mistake at the time of program replacement.*" Thus, Nishiyama is concerned with managing software upgrades in a network, not servicing user requests issued to a server. The applicants respectfully submit that Nishiyama does ***not*** "show substantial features of the claimed invention," as asserted in

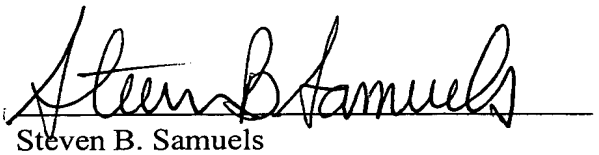
the Official Action. Nor does either Wischinski, Bryan or Mutschler, the secondary references relied upon in the Official Action.

Because Nishiyama merely describes a software version/revision management system, which has nothing to do with the concept recited in independent claims 1, 13 and 14 of the instant application, the applicants respectfully submit that those claims patentably define over Nishiyama, whether alone or in combination with Wischinski, Bryan, Mutschler or any other cited art of record. Moreover, inasmuch as claims 2-12 and 15-18 each depend either directly or indirectly from one of the independent claims, the applicants submit that they too patentably define over the art of record for the same reasons. Reconsideration of the Section 103(a) rejection of claims 1-18 is therefore respectfully requested.

CONCLUSION

For all the foregoing reasons, the applicants respectfully submit that the present application is in condition for allowance.

Respectfully submitted,



Date: March 5, 2007

Steven B. Samuels
Registration No. 37,711

Woodcock Washburn LLP
Cira Centre
2929 Arch Street, 12th Floor
Philadelphia, PA 19104-2891
Telephone: (215) 568-3100
Facsimile: (215) 568-3439

Please address all correspondence to:
Mark T. Starr, Esq.
Patent Counsel
Unisys Corporation
Unisys Way
Blue Bell, PA 19424
Telephone: (215) 986-4411
Facsimile: (215) 986-3090